软件项目风险管理

一、风险管理概述

软件风险是指软件开发过程中及软件产品本身可能造成的伤害或损失。风险关注未来的事情，这意味着，风险涉及选择及选择本身包含的不确定性，在软件开发过程及软件产品都要面临各种决策的选择。风险是介于确定性和不确定性之间的状态，是处于无知和完整知识之间的状态。另一方面，风险将涉及思想、观念、行为、地点等因素的改变。

当在软件工程领域考虑风险时，我们要关注以下的问题：什么样的风险会导致软件项目的彻底失败？用户需求、开发技术、目标计算机、以及所有其它与项目有关的因素的改变将会对按时交付和总体成功产生什么影响？对于采用什么方法和工具，需要多少人员参与工作的问题，我们如何选择和决策？对软件质量要达到什么程度才是“足够的”？

当没有办法消除风险，甚至连试图降低该风险也存在疑问时，这些风险就是真正的风险了。在我们能够标识出软件项目中的真正风险之前，识别出所有对管理者和开发者而言均为明显得风险是很重要的。

二、被动和主动的风险策略

被动风险策略是针对可能发生的风险来监督项目，直到它们变成真正的问题时，才会拨出资源来处理它们，更普遍的是，软件项目组对风险不闻不问，直到发生了错误才赶紧采取行动，试图迅速地纠正错误。这种管理模式常常被称为“救火模式”。当补救的努力失败后，项目就处在真正的危机之中了。

对于风险管理的一个更聪明的策略是主动式的。主动策略早在技术工作开始之前就已经启动了——标识出潜在地风险，评估它们出现的概率及产生的影响，对风险按重要性进行排序，然后，软件项目组建立一个计划来管理风险。主动策略风险管理的主要目标是预防风险。但是，因为不是所有的风险都能够预防，所以，项目组必须建立一个应付意外事件的计划，使其在必要时能够以可控的及有效的方式作出反应。

三、软件风险

1、软件风险包含两个特征：

不确定性 —— 刻划风险的事件可能发生也可能不发生，没有100 %发生的风险。

损失 —— 如果风险变成了现实，就会产生恶性后果或损失。

2、进行风险分析时，重要的是量化不确定的程度和与每个风险相关的损失的程度。

为了实现这点，必须考虑以下几种不同类型的风险：

项目风险 ：项目风险是指潜在的预算、进度、人力（工作人员和组织）、资源、客户、需求等方面的问题以及它们对软件项目的影响。项目风险威胁项目计划，如果风险变成现实，有可能会拖延项目的进度，增加项目的成本。项目风险的因素还包括项目的复杂性、规模、结构的不确定性。

技术风险 ：是指潜在地设计、实现、接口、验证和维护等方面的问题。此外规约的二义性、技术的不确定性、陈旧的技术、以及 "过于先进"的技术也是风险因素。技术风险威胁要开发的软件的质量及交付时间。如果技术风险变成现实，则开发工作可能变得很困难或者不可能。

商业风险 ：商业风险威胁到要开发软件的生存能力。商业风险常常会危害项目或产品。

五个主要的商业风险是：

（1）开发一个没有人真正需要的优秀产品或系统（市场风险）；

（2）开发的产品不再符合公司的整体商业策略（策略风险）；

（3）建造了一个销售部门不知道如何去卖的产品；

（4）由于重点的转移或人员的变动而失去了高级管理层的支持（管理风险）；

（5）没有得到预算或人力上的保证（预算风险）。

3、风险分为以下方式：

（1）已知风险 ,是通过仔细评估项目计划、开发项目的商业及技术环境、以及其它可靠的信息来源（如：不现实的交付时间，没有需求或软件范围的文档、恶劣的开发环境）之后可以发现的那些风险。

（2）可预测风险 ，能够从过去项目的经验中推测出来（如：人员调整，与客户之间无法沟通，由于需要进行维护而使开发人员精力分散）。

（3）不可预测风险 ，它们可能、也会真的出现，但很难事先识别出它们来。

四、识别风险

识别风险是试图系统化地确定对项目计划（估算、进度、资源分配）的威胁。通过识别已知和可预测的风险，项目管理者就有可能避免这些风险，且当必要时控制这些风险。

每一类风险可以分为两种不同的类型：一般性风险和特定产品的风险。一般性风险对每一个软件项目而言都是一个潜在地威胁。特定产品的风险只有那些对当前项目的技术、人员、及环境非常了解的人才能识别出来。为了识别特定产品的风险，必须检查项目计划及软件范围说明，从而了解本项目中有什么特殊的特性可能会威胁到项目计划。

一般性风险和特定产品的风险都应该被系统化地标识出来。 识别风险的一个方法是建立风险条目检查表。该检查表可以用来识别风险，并可以集中来识别下列常见子类型中已知的及可预测的风险：

- 产品规模 —— 与要建造或要修改的软件的总体规模相关的风险。
- 商业影响 —— 与管理或市场所加诸的约束相关的风险。
- 客户特性 —— 与客户的素质以及开发者和客户定期通信的能力相关的风险。
- 过程定义 —— 与软件过程被定义的程度以及它们被开发组织所遵守的程度相关的风险。
- 开发环境 —— 与用以建造产品的工具的可用性及质量相关的风险。
- 建造的技术 —— 与待开发软件的复杂性以及系统所包含技术的 "新奇性"相关的风险。
- 人员数目及经验 —— 与参与工作的软件工程师的总体技术水平及项目经验相关的风险。

风险条目检查表能够以不同的方式来组织。与上述话题相关的问题可以由每一个软件项目来回答。这些问题的答案使得计划者能够估算风险产生的影响。

1、产品规模风险

项目风险是直接与产品规模成正比的。下面的风险检查表中的条目标识了产品（软件）规模相关的常见风险：

- 是否以 LOC 或 FP 估算产品的规模；
- 对于估算出的产品规模的信任程度如何；
- 是否以程序、文件或事务处理的数目来估算产品规模；
- 产品规模与以前产品的规模的平均值的偏差百分比是多少；
- 产品创建或使用的数据库大小如何；
- 产品的用户数有多少；
- 产品的需求改变多少？交付之前有多少？交付之后有多少？
- 复用的软件有多少？

2、商业影响风险

销售部门是受商业驱动的，而商业考虑有时会直接与技术实现发生冲突。下面的风险检查表中的条目标识了与商业影响相关的常见风险：

- 本产品对公司的收入有何影响；
- 本公司是否得到公司高级管理层的重视；
- 交付期限的合理性如何；
- 将会使用本产品的用户数及本产品是否与用户的需要相符合；
- 本产品必须能与之互操作的其它产品 /系统的数目；
- 最终用户的水平如何；
- 政府对本产品开发的约束；
- 延迟交付所造成的成本消耗是多少；
- 产品缺陷所造成的成本消耗是多少；

对于待开发产品的每一个回答都必须与过去的经验加以比较。 如果出现了较大的百分比偏差或者如果数字接近过去很不令人满意的结果，则风险较高。

3、客户相关风险

客户有不同的需要。一些人只知道他们需要什么；而另一些人知道他们不需要什么。一些客户希望进行详细的讨论，而另客户则满足于模糊的承诺。

客户有不同的个性。一些人喜欢享受客户的身份，而另一些人则根本不喜欢做为客户。一些人会高兴地接受几乎任何交付的产品，并能充分利用一个不好的产品；而另一些人则会对质量差的产品猛烈抨击。一些人会对质量好的产品表示赞赏；而另一些人则不管怎样都抱怨不休。

客户和供应商之间也有各种不同的通信方式。一些人非常熟悉产品及生产厂商；而另一些人则可能素未谋面，仅仅通过信件来往和电话与生产厂商沟通。

一个“不好的”客户可能会对一个软件项目组能否在预算内完成项目产生很大的影响。对于项目管理者而言，不好的客户是对项目计划的巨大威胁和实际的风险。下面的风险检查表中的条目标识了与客户特征相关的常见风险：

- 你以前是否曾与这个客户合作过；
- 该客户是否很清楚需要什么；他能否化时间把需求写出来；
- 该客户是否同意花时间召开正式的需求收集会议，以确定项目范围；
- 该客户是否愿意建立与开发者之间的快速通信渠道；
- 该客户是否愿意参加复审工作；
- 该客户是否具有改产品领域的技术素养；
- 该客户是否愿意你的人来做他们的工作；
- 该客户是否了解软件过程；

如果对于这些问题中的任何一个答案是否定的，则需要进一步的调研，以评估潜在地风险。

4、过程风险

如果软件过程定义得不清楚；如果分析、设计、测试以无序的方式进行；如果质量是每个人都认为很重要的概念，但没有人切实采取行动来保证它，那么这个项目就处在风险之中。

过程问题：

- 高级管理层是否有一份已经写好的政策陈述，该陈述中强调了软件开发标准过程的重要性；
- 开发组织是否已经拟定了一份已经成文的、用于本项目开发的软件过程的说明；
- 开发人员是否同意按照文档所写的软件过程进行开发工作，并自愿使用它；
- 该软件过程是否可以用于其它项目；
- 管理者和开发人员是否接受过一系列的软件工程培训；
- 是否为每一个软件开发者和管理者提供了印好的软件工程标准；
- 是否为作为软件过程一部分而定义的所有交付物建立了文档概要及示例；
- 是否定期对需求规约、设计和编码进行正式的技术复审；
- 是否定期对测试过程和测试情况进行复审；
- 是否对每一次正式技术复审的结果建立了文档，其中包括发现的错误及使用的资源；
- 有什么机制来保证按照软件工程标准来指导工作；
- 是否使用配置管理来维护系统　/软件需求、设计、编码、测试用例之间的一致性；
- 是否使用一个机制来控制用户需求的变化及其对软件的影响；
- 对于每一个承包出去的子合同，是否有一份文档化的工作说明、一份软件需求规约和一份软件开发计划；
- 是否有一个可遵循的规程，来跟踪及复审子合同承包商的工作；

技术问题

- 是否使用方便易用的规格说明技术来辅助客户与开发者之间的通信；

- 是否使用特定的方法进行软件分析；
- 是否使用特定的方法进行数据和体系结构的设计；
- 是否 90 %以上的代码都是使用高级语言编写的；
- 是否定义及使用特定的规则进行代码编写；
- 是否使用特定的方法进行测试用例的设计；
- 是否使用配置管理软件工具控制和跟踪软件过程中的变化活动；
- 是否使用工具来创造软件原型；
- 是否使用软件工具来支持测试过程；
- 是否使用软件工具来支持文档的生成和管理；
- 是否收集所有软件项目的质量度量值；
- 是否收集所有软件项目的生产率度量值；

如果对于上述问题的答案多数是否定的，则软件过程是薄弱的，且风险很高

5、技术风险

突破技术的极限极具挑战性和令人兴奋，但这也是有风险的。下面的风险检查表中的条目标识了与建造的技术相关的常见风险：

- 该技术对于你的公司而言是新的吗；
- 客户的需求是否需要创建新的算法或输入、输出技术；
- 待开发的软件是否需要使用新的或未经证实的硬件接口；
- 待开发的软件是否需要与开发商提供的未经证实的软件产品接口；
- 待开发的软件是否需要与功能和性能均未在本领域得到证实的数据库系统接口；
- 产品的需求是否要求采用特定的用户界面；
- 产品的需求中是否要求开发某些程序构件，这些构件与你的公司以前开发的构件完全不同；
- 需求中是否要求采用新的分析、设计、测试方法；
- 需求中是否要求使用非传统的软件开发方法；
- 需求中是否有过分的对产品的性能约束；
- 客户能确定所要求的功能是可行的吗？

如果对于这些问题中的任何一个答案是肯定的，则需要进一步的调研，以评估潜在地风险。

6、开发环境风险

软件工程环境支持项目组、过程及产品，但是，如果环境有缺陷，它就有可能成为重要的风险源。下面的风险检查表中的条码标识了与开发环境相关的风险：

- 是否有可用的软件项目管理工具；
- 是否有可用的软件过程管理工具；
- 是否有可用的分析及设计工具；
- 分析和设计工具是否适用于待建造产品；
- 是否有可用的编译器或代码生成器；

- 是否有可用的测试工具；
- 是否有可用的软件配置管理工具；
- 环境是否利用了数据库或数据仓库；
- 项目组的成员是否接受过每个所使用工具的培训；
- 是否有专家能够回答有关工具的问题　　；
- 工具的联机帮助及文档是否适当；

如果对于上述问题的答案多数是否定的，则软件开发环境是薄弱的，且风险很高。

7、与人员数目及经验相关的风险

- 是否有最优秀的人员可用；
- 人员在技术上是否配套；
- 是否有足够的人员可用；
- 开发人员是否能够自始至终地参加整个项目的工作；
- 项目中是否有一些人员只能部分时间工作；
- 开发人员对自己的工作是否有正确的期望；
- 开发人员是否接受过必要的培训；
- 开发人员的流动是否仍能保证工作的连续性；

如果对于这些问题中的任何一个答案是否定的，则需要进一步的调研，以评估潜在地风险。

8、风险因素和驱动因子

为了很好地识别和消除软件风险，项目管理者需要标识影响软件风险因素的风险驱动因子，这些因素包括性能、成本、支持和进度。风险因素是以如下的方式定义的：

- 性能风险 —— 产品能够满足需求且符合于其使用目的的不确定的程度。
- 成本风险 —— 项目预算能够被维持的不确定的程度。
- 支持风险 —— 软件易于纠错、适应及增强的不确定的程度。
- 进度风险 —— 项目进度能够被维持且产品能按时交付的不确定的程度。

每一个风险驱动因子对风险因素的影响均可分为四个影响类别　　—— 可忽略的、轻微的、严重的、灾难性的。下表指出了由于错误而产生的潜在影响或没有达到预期的结果所产生的潜在影响。影响类别的选择是以最符合表中描述的特性为基础的。

影响评估					
类别＼因素		性　能	支　持	成　本	进　度
灾难的	1	无法满足需求而导致任务失败		错误将导致进度延迟和成本增加	
	2	严重退化使得根本无法达到要求的技术性能	无法作出响应或无法支持的软件	严重的资金短缺，很可能超出预算	无法在交付日期内完成

严重的	1	无法满足需求而导致系统性能下降，使得任务能否成功受到置疑		错误将导致操作的延迟，并使成本增加	
	2	技术性能有所下降	在软件修改中有少量的延迟	资金不足，可能会超支	交付日期可能延迟
轻微的	1	无法满足要求而导致次要任务的退化		成本、影响和即可恢复的进度上的小问题	
	2	技术性能有较小的降低	较好的软件支持	有充足的资金来源	实际的、可完成的进度计划
可忽略的	1	无法满足要求而导致使用不方便或不易操作		错误对进度及成本的影响很小	
	2	技术性能不会降低	易于进行软件支持	可能低于预算	交付日期将会提前
注： 1、未测试出的软件错误或缺陷所产生的潜在影响。 2、如果没有达到预期的结果所产生的潜在影响。					

五、风险预测

风险预测，又称风险估算，试图从两个方面评估每一个风险 —— 风险发生的可能性或概率，以及风险发生了，所产生的后果。项目计划者、其它管理人员和技术人员一起执行四个风险预测活动：

（1）建立一个尺度，以反映风险发生的可能性；

（2）描述风险的后果；

（3）估算风险对项目及产品的影响；

（4）标注风险预测的整体精确度，以免产生误解。

1、建立风险表

风险表给项目管理者提供了一种简单的风险预测技术。（样本如下表）

项目组一开始要在表中的第一列列出所有风险可能，这些可以利用前面所述的风险检查条目来完成。在第二列队风险进行分类，风险发生概率放在第三列。每个风险的概率值可以由项目组成员个别估算，然后将这些值平均，得到一个有代表性的概率值。

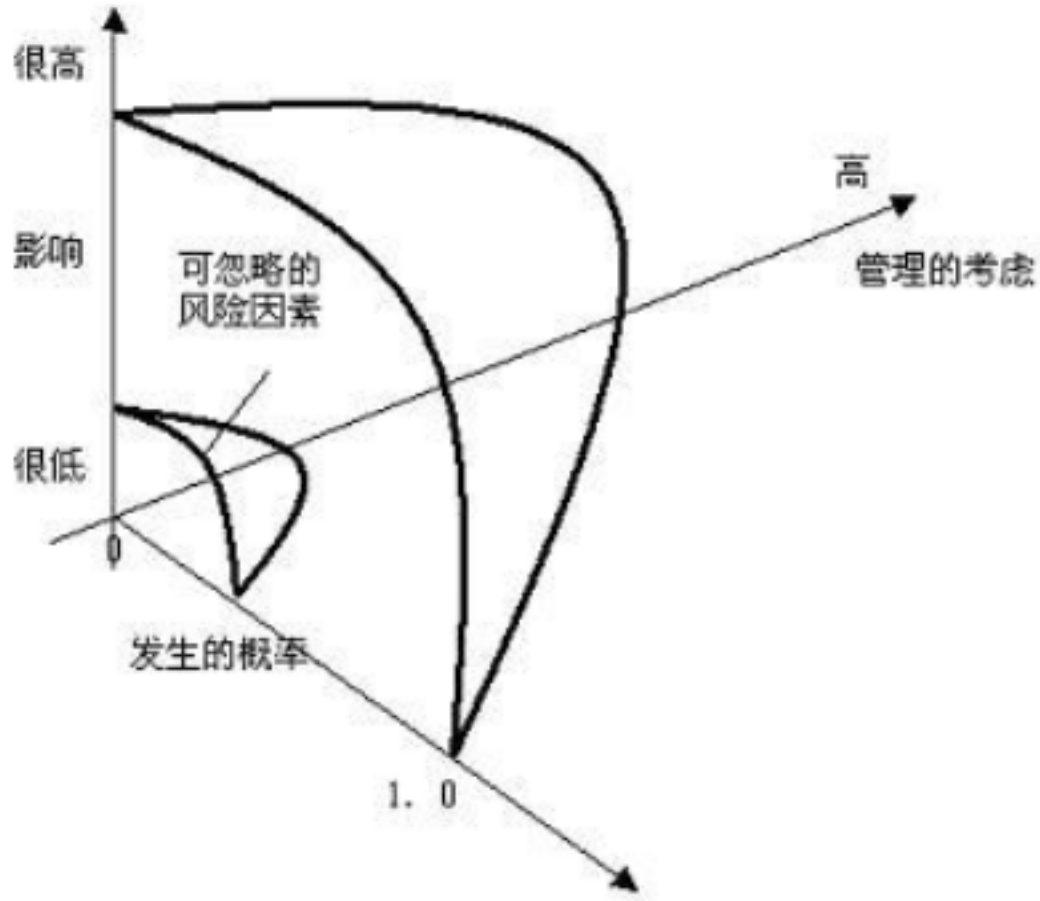
分类前的风险表样本				
风险	类别	概率	影响	RMMM
规模估算可能非常低	PS	60%	2	
用户数量大大超出计划	PS	30%	3	
复用程度低于计划	PS	70%	2	
最终用户抵制该计划	BU	40%	3	
交付期限将被紧缩	BU	50%	2	
资金将回流失	CU	40%	1	

用户将改变需求	PS	80%	2	
技术达不到预期的效果	TE	30%	1	
缺少对工具的培训	DE	80%	3	
人员缺乏经验	ST	30%	2	
人员流动频繁	ST	60%	2	
·				
·				
注：影响类别取值： 1—灾难的 2—严重的 3—轻微的 4—可忽略的				

一旦完成风险表的前四列内容，就要根据概率及影响来进行排序。高概率、高影响的风险放在表的上方。这就完成了第一次风险排序。

项目管理者研究已经排序的表，并定义一条终止线。该终止线（表中某一点上的一条水平线）表示：只有在那些线上的风险才会得到进一步的关注，线之下的风险则需要再评估以完成第二次排序。

风险影响及概率从管理的角度来考虑，是起着不同作用的（见下图）。一个具有高影响但发生概率很低的风险因素不应该花费太多的管理时间。而高影响且发生概率为中到高的风险以及低影响但高概率的风险，应该首先考虑。



2、评估风险影响

如果风险真的发生了，所产生的后果有三个因素可能会受影响：风险的性质、范围、时间。风险的性质是指当风险发生时可能产生的问题。例如，一个定义得很差的与客户硬件的接口（技术风险）会妨碍早期的设计和测试，也有可能导致项目后期阶段的系统集成问题。风险的范围结合了严重性及其整体分布情

况。风险的时间主要考虑何时能够感到风险，风险会持续多长时间。在大多数情况下，项目管理者希望 “坏消息”越早出现越好。

以下的步骤用来确定风险的整体影响：

- 确定每个风险元素发生的平均概率。
- 使用前面的表格，基于其中列出的标准来确定每个因素的影响。
- 完成风险表，分析其结果。
- 风险预测和分析技术可以在软件项目进展过程中跌代使用。项目组定期复查风险表，再评估每一个风险，以确定新的情况是否引起其概率及影响的改变。

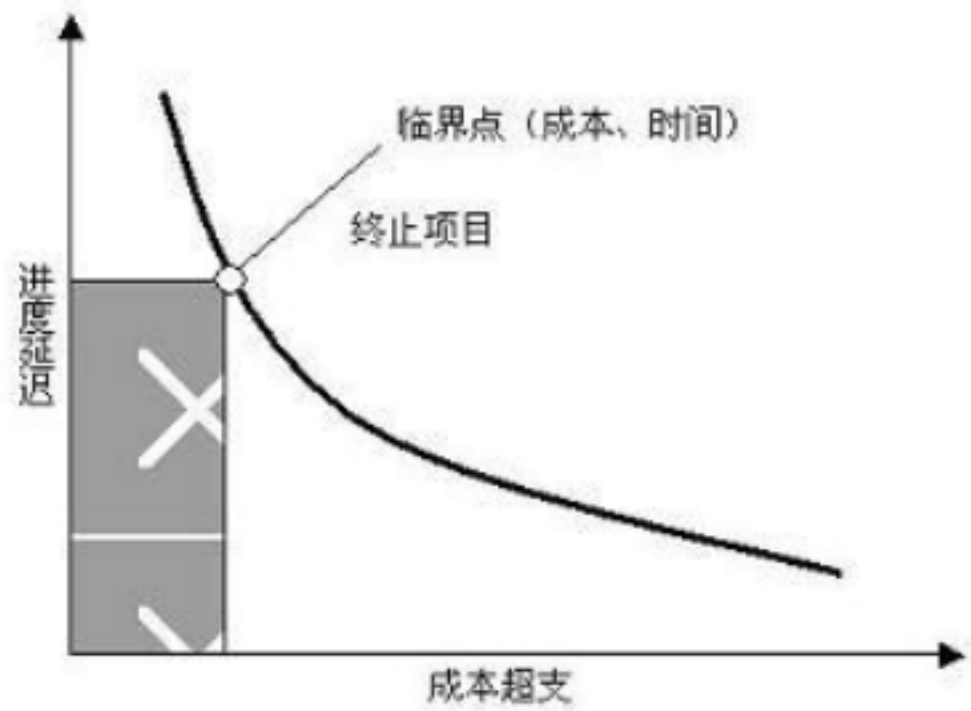
3、风险评估

我们建立如下形式的一系列三元组： [r,l,x]

其中 r 表示风险， l 表示风险发生的概率， x 表示风险产生的影响。在风险评估过程中，我们进一步审查在风险预测阶段所做的估算的精确度，试图为所发现的风险排出优先次序，并开始考虑如何控制或避免可能发生的风险。

要使评估发生作用，必须定义一个风险参考水平值。对于大多数软件项目而言，前面讨论的风险因素——性能、成本、支持、进度，也代表了风险参考水平值。即，对于性能下降、成本超支、支持困难、或进度延迟，都有一个水平值的要求，超过它就会导致项目被迫停止。如果风险的组合所产生的问题引起一个或多个参考水平值被超过，则工作将会停止。在软件风险分析中，风险参考水平值存在一个点，称为参考点或临界点，在这个点上，决定继续进行该项目或终止它（问题太多）都是可以接受的。

下图以图形方式表示了这种情况。如果风险的组合产生问题导致成本超支及进度延迟，则会有一个水平值 ,即图中的曲线，当超过它时会引起项目终止。



实际上， 参考水平很少能表示成光滑曲线。 在大多数情况下， 它是一个区域， 其中存在很多不确定性。

因此，在风险评估中，我们执行以下步骤：

- 定义项目的风险参考水平值；

- 建立每一组 [r,l,x] 与每一个参考水平值之间的关系；
- 预测一组临界点以定义项目终止区域，该区域由一条曲线或不确定区域界定。
- 预测什么样的风险组合会影响参考水平值。

六、风险缓解、监控和管理

进一步的所有风险分析活动都只有一个目的 —— 辅助项目组建立处理风险的策略。 一个有效的策略必须考虑三个问题：

- 风险避免
- 风险监控
- 风险管理及意外事件计划

如果软件项目组对于风险采取主动的方法，则避免永远是最好的策略。这可以通过建立一个风险缓解计划来达到。例如，频繁的人员流动被标注为一个项目风险，基于以往的历史和管理经验，人员流动的概率为 70％，而影响被预测卫对于项目成本及进度有严重的影响。为了缓解这个风险，项目管理者必须建立一个策略来降低人员流动。可能采取的策略如下：

- 与现有人员一起探讨一下人员流动的原因（如恶劣的工作条件，低报酬，竞争激烈）
- 在项目开始之前，采取行动以缓解那些在管理控制之下的原因。
- 一旦项目启动，假设会发生人员流动并采取一些技术措施以保证当人员离开时的工作连续性。
- 对项目进行良好组织，使得每一个开发活动的信息能被广泛传播和交流。
- 定义文档的标准，并建立相应的机制，以确保文档能被及时建立。
- 对所有工作进行详细复审，使得不止一个人熟悉该项工作。
- 对于每一个关键的技术人员都指定一个后备人员。

随着项目的进展，风险监控活动开始进行。项目管理者监控某些因素，这些因素可以提供风险是否正在变高或变低的指示。在上例中，应该监控下列因素：

- 项目组成员对项目压力的一般态度。
- 项目组的凝聚力。
- 项目组成员彼此之间的关系。
- 与报酬和利益相关的潜在问题
- 在公司内及公司外工作的可能性。

除了监控上述因素之外，项目管理者还应该监控风险缓解步骤的效力。例如：上例中，风险缓解步骤要求定义 “文档的标准，并建立相应的机制，以确保文档能被及时建立 ”。如果有关键的人物离开了项目组，这是保证工作连续性的机制。项目管理者应该仔细地监控这些文档，以保证文档内容正确，当新员工加入该项目时，能为他们提供必要的信息。

风险管理及意外事件计划假设缓解工作已经失败，风险变成了现实。继续前面的例子，假定项目正在进行中，有一些人宣布将要离开。如果按照缓解策略行事，则有后备人员可用，因为信息已经文档化，有

关知识已经在项目组中广泛进行了交流。此外，项目管理者还可以暂时重新将资源调整到那些需要人的地方去，并调整项目进度，从而使新加入的成员能够 “赶上进度”。同时，要求那些要离开的人员停止工作，进入“知识交接模式”。

RMMM 步骤将导致额外的项目开销。 因此，风险管理的部分任务是评估何时由 RMMM 步骤所产生的效益低于实现它们所花费的成本。本质上是讲，项目计划者执行一个典型的成本－效益分析来估算项目开销变化情况。

对于一个大型项目，可能会标识出 30－40 种风险。如果为每种风险定义三至七个风险管理步骤，则风险管理本身就可能变成一个 “项目”。经验表明：整个软件风险的 80％（即可能导致项目失败的 80％潜在的因素）能够由仅仅 20％的已知风险来说明。早期风险分析步骤中所实现的工作能够帮助计划者确定哪些风险在所说的 20％中。

1、安全性风险和危险

风险不仅限于软件项目本身。在软件已经能够交付客户之后，仍有可能发生风险。这些风险一般与领域中的软件失败相关。

虽然一个良好的系统发生错误的概率很小， 但是基于计算机的控制及监督系统中未被发现的错误可能会导致巨大的经济损失，或者更加严重

当软件被用作控制系统的一部分时， 复杂性会以数量级增加。 由于人的错误所引起的微小的设计缺陷，在使用软件时会变得难以发现。

软件安全和危险分析是属于软件质量保证活动， 它主要是用来标识和评估可能对软件产生负面影响并使整个系统失败的潜在危险。如果能够在软件工程的早期阶段标出危险，则可以指定软件设计特征来消除或控制潜在地危险。

2、 RMMM 计划

风险管理策略可以包含在软件项目计划中，或者风险管理步骤也可以组织成一个独立的风险缓解、监控和管理计划（ RMMM 计划）。 RMMM 计划将所有风险分析文档化，并由项目管理者作为整个项目计划中的一部分来使用。 RMMM 计划的大纲如下：

Ⅰ.引言
文档的范围和目的
主要风险综述
责任
a.管理者
b.技术人员

Ⅱ.项目风险表
终止线之上所有风险的描述
影响概率及影响的因素

Ⅲ.风险缓解、监控和管理

缓解

一般策略

缓解风险的特定步骤

监控

被监控的因素

监控办法

管理

意外事件计划

特殊的考虑

Ⅳ.RMMM 计划的迭代时间安排表

Ⅴ总结

七、软件风险的总结

当对软件项目期望值很高时，一般都会进行风险分析。不过，即使进行这项工作，大多数软件管理者都是非正式地和表面地完成它。化在标识、分析、管理风险上的时间可以从多个方面得到回报：更加平稳的项目进展过程；较高的跟踪和控制项目的能力；因为周密计划而产生的信心。